INTRODUCTION

The aim of this study is improve my knowledge about PostgreSQL language and it’s syntax.

DETAILS

SELECT

SELECT first\_name || ‘ ‘ || lastname FROM Customer ; : Writes the firstname and lastname columns adjacent to a single column. (as Faruk Koyuncu)

SELECET DISTINCT first\_name From Customers ; : Retrieve only unique values from the first\_name column

SELECET DISTINCT first\_name, last\_name From Customers ; : In this case, the combination of values in both column1 and column2 columns will be used for evaluating the duplicate. (Data can be : Faruk kırmızı

Faruk beyaz)

SELECET DISTINCT ON(first\_name) first\_name, last\_name From Customers ; : Looks only for first\_name to be unique, last\_name values can be duplicate.

AS

SELECT first\_name isimler, last\_name FROM Customer ; : When we use Aliases, we don’t have to use AS keyword.

WHERE

SELECT first\_name, last\_name FROM Customer WHERE first\_name = ‘Faruk’

SELECT first\_name, last\_name FROM Customer WHERE first\_name IN ('Ann','Anne','Annie'); : Retrieve the datas of the customer whose First name is Ann, Anne, Annie.

SELECT first\_name, last\_name FROM Customer WHERE first\_name LIKE 'Ann%'; : In this case, retrieved customers first\_name must start with ‘Ann’.

SELECT first\_name, last\_name FROM Customer WHERE first\_name LIKE 'A%' AND LENGTH(first\_name) BETWEEN 3 AND 5 ORDER BY name\_length; : We can use BETWEEN keyword like this.

JOINS

SELECT a.first\_name, b.money\_amount FROM Customers a, Tenants b WHERE a.customer\_id = b.c\_id ; : Retrieves the first name and Money amount from the customers and tenants table by matching them with their customer id in two tables.

UNIONS

To combine the result sets of two queries using the UNION operator, the queries must conform to the following rules:

* The number and the order of the columns in the select list of both queries must be the same.
* The data types must be compatible.

SELECT \* FROM top\_rated\_films UNION SELECT \* FROM most\_popular\_films; : The following statement uses the UNION operator to combine data from both tables. Duplicates are removed with Union operator.

SELECT \* FROM top\_rated\_films UNION ALL SELECT \* FROM most\_popular\_films; : In this version of Union, duplicate rows are not removed from the retrieved data.

UPDATE

UPDATE courses SET published\_date = '2020-07-01' WHERE course\_id = 2 RETURNING \*; : Updates the stated data and returning the updated value.

UPDATE product SET net\_price = price - price \* discount FROM product\_segment WHERE product.segment\_id = product\_segment.id; : In this case we do UPDATE with JOIN. Suppose you have to calculate the net price of every product based on the discount of the product segment. To do this, you can apply the UPDATE join statement.

DELETE

DELETE FROM links WHERE id = 7 RETURNING \*; : DELETE operations with Where Condition.

DELETE FROM links WHERE id IN (6,5) RETURNING \*;

DELETE FROM contacts USING blacklist WHERE contacts.phone = blacklist.phone; : Delete Operation with JOIN.

REFERENCES

[https://www.postgresqltutorial.com/postgresql-tutorial/\*](https://www.postgresqltutorial.com/postgresql-tutorial/*) : The star indicator means every subdomain in this Tutorial website.